Affect Product
Tuya Smart Life App (version 5.6.1), and all vendor apps using Tuya Smart IoT SDK, like ledvance, lloyds Smart, uHome+, Hama Smart.

Threat Model

In this report, we consider the scenario where the usage rights of the device are shared with another user. Such sharing is common in the IoT lifecycle. For example, in hotels/Airbnb/apartments, a guest/tenant is granted the right to use a device during their stay, and the owner/administrator revokes the permissions of ex-users later (e.g. when they leave/check out). We consider that the ex-user may intend to monitor/control the device even after her access to the device is revoked.

Background

1. Matter provides open-source development tools [1] such as CHIP Tool, which offers a comprehensive set of Matter Controller functionalities to operate Matter devices.
2. Using the CHIP Tool as a Matter controller allows for the free modification of its Vendor ID and Label values. Example command:
   ./chip-tool pairing code [Node ID] [Pairing Code] --commissioner-vendor-id [Vendor ID]
   ./chip-tool operationalcredentials update-fabric-label [Label Value] [Node ID] [Endpoint]
3. In mobile Apps of major vendors, homes are primarily used as the basis for displaying related devices. When users want to add devices, they must first create a home, thereby establishing a relationship between the user and the home. Following network configuration, a relationship is then established between the device and the home, leading to a relationship between the user and the device. Tuya thinks that Matter devices are no exception; if they are to be used within an app, the common practice is to create a one-to-one fabric for the home. Subsequently, within the fabric corresponding to the home, devices are assigned specific node IDs, facilitating the integration of Matter devices.

Attack

The Security Weaknesses

1. An attacker with only normal-guest-level permissions can unexpectedly invoke some APIs related to the Matter protocol (e.g. issue the credentials), which is not allowed in the UI of the Tuya App. Subsequently, the attacker can leverage these APIs to issue certificates within the Matter protocol for themselves and obtain all the related parameters for controllers of the fabric and devices. With these parameters, the

attacker can use the CHIP Tool to gain control of the fabric, thereby assuming the role of a Matter administrator.
2. By the design of the Matter protocol, the fabric administrator can add arbitrary devices and name them with arbitrary Node IDs. Besides, when two physical devices have the same Node ID, the Matter controller in the fabric will be confused about choosing the correct one, leading to misoperations (eg., the device owner tries to control her/his Matter device, but her/his device has no response since she/he indeed controls the malicious device with the same Node ID).
3. In the Tuya App, the guest is over-privileged in Matter devices' ACL. Specifically, in our test of the device Zemismart LED E27, its ACL records 5 (Admin Level) both for the device owner and the guest.
4. In the Tuya App, the Node IDs of users (owner, guests) are incremental, allowing guests to enumerate the Node ID of the device owner and impersonate the owner.
5. In the Tuya App, the Node IDs of devices in the same home are incremental too, allowing a temporary guest to predict the next device's Node ID even if their permission was revoked by the owner.

Attack A (Convert control channel)

Proof-of-Concept Attacks:

By exploiting the weakness above, we describe the practical attacks with technical details as follows:

1. The device owner invites a guest (as a malicious attacker) into his Tuya home and grants the guest only "Normal User" permissions who can only use the device.
2. Through reverse engineering with the Tuya mobile app, the attacker can hook their app to interact with the Tuya cloud through some APIs.
3. The attacker can initiate a request to the *thing.m.matter.icac.issue* API, including the Tuya home's group ID (gid), and an ECC public key to be signed. The response includes RCAC and ICAC certificates signed by the Tuya Cloud, the fabric ID of the current home, and the node ID assigned to the attacker within that fabric. Every node within a fabric possesses the same root certificate, directly correlating the certificates issued by the thing.m.matter.icac.issue API to the fabric of the respective home.
4. Additionally, by requesting the thing.m.matter.device.node.batch.get API, the node IDs of devices within the fabric in the home can be obtained.
5. The attacker can utilize the CHIP Tool with the fabric certificates, the fabric ID, and node IDs obtained in steps 3 and 4 to become the fabric admin.
6. The device owner subsequently revokes the guest's access by removing them from the home.
7. By leveraging weakness 3, the temporal guest can still control the device using CHIP-Tool with their own Node ID.

The attack video is at [3].

Also, see a simpler attack at [8] by utilizing Tuya API *thing.m.matter.derivation.passcode.get,* and hook the MultipleFabricPasscode function to open the device's commissioning window out of the device owner's expectations.

Attack A.1 (Device-Control-Device)

Continues with steps 1-7 in Attack 2.A, the attack impact can be extended with additional steps.

8. The attacker can then add a malicious device(a switch) with Node ID *X* into the owner's fabric.
9. The attacker binds the malicious switch with the benign device using Matter protocol cluster *Bind*, such that when the switch is pressed, the switch tries to toggle the benign device.
10. By leveraging weakness 3, the attacker writes a new ACL to the device as follows using the Matter protocol cluster *AccessControl*, granting the switch the highest privilege level (5).
    [{ "fabricIndex" : 1 , "privilege" : 5 , "authMode" : 2 , "subjects" : [65536,65537,X] , "targets" : null }]

11. Now the attacker can control the device by physically operating his malicious switch. Notably, the malicious switch is not shown in the owners' Tuya App.

Please see the attack video at [6].

Attack A.2 (DOS by ACL modifying)

Continues with steps 1-7 in Attack 2.A, the attack impact can be extended with additional steps.

8. The attacker uses CHIP-Tool to modify the device' ACL, decreasing the privilege level of the owner.
9. The owner can not control the device or open the device commission window. And the App does not give any clues or explanations.

Please see the attack video at [4].

Attack B

1-6. The same with Attack 2.A

7. By leveraging weakness 4. the temporal guest can obtain the correct Node ID of the device owner by brute-forcing starting from his Node ID. Then he can still control the device using CHIP-Tool with the owner's Node ID, impersonating the device owner.

Attack C (DOS by device's Node ID hijacking)

1-6. The same with Attack 2.A

7. By leveraging weakness 5, the attacker predicts the next device‘s Node ID *X*.
8. The attacker adds his malicious device (which can be a script like matter.js []) with Node ID *X* to the fabric.
9. The owner adds a benign device to the fabric with the Tuya App, automatically assigning the Node ID *X* to it.
10. The owner attempts to control the benign device with the Tuya App, but the device has no response since the commands are incorrectly sent to the malicious device with the same Node ID when the benign device fails the competition.

See the attack video at [7].

Attack D

1-6. The same with Attack 2.A

7. Just like Attack 2.C, even without weakness 5, as long as there is already a benign device in the home when the owner invites the guest, the guest can obtain the device’s Node ID *X* in step 4,
8. The guest adds his malicious device with Node ID *X* to the fabric.
9. The owner attempts to control the benign device with the Tuya App, but the device has no response since the commands are incorrectly sent to the malicious device with the same Node ID when the benign device fails the competition.

Reference

[1] https://github.com/project-chip/connectedhomeip
[2] https://github.com/project-chip/matter.js
[3] https://drive.google.com/file/d/1bSLnr8HVqRBJWpULePDGoFvJB1tqBhpY/view?usp=share_link
[4] https://drive.google.com/file/d/1OeDdAng9iXOQTdZEcBhEZrQ9cBQrCDVI/view?usp=share_link
[5] https://drive.google.com/file/d/1dGDVA0gfvrljE7w97Iv5B_EjHoPkQTO1/view?usp=share_link
[6] https://drive.google.com/file/d/1Gbg9U8ZwGtNh9x6goQQ0SbmqFF7oFZ2z/view?usp=share_link
[7]https://drive.google.com/file/d/1__-K7WMbYXQilU35lOr48_fyVfSHPy30/view?usp=sharing

[8]https://drive.google.com/file/d/1iYDetul36zeyTzsd85pJIKT4IL7NNVGJ/view?usp=drive_link